An Inconsistency Problem in Data Discretion Using Equal Width Interval Approach (Mathematical Programming in the 21st Century : Algorithms and Modeling)

AUTHOR(S):

Wu, Chien-Hsing; Ishii, Hiroaki

CITATION:

Wu, Chien-Hsing ...[et al]. An Inconsistency Problem in Data Discretion Using Equal Width Interval Approach (Mathematical Programming in the 21st Century : Algorithms and Modeling). 数理解析研究所講究録 2010, 1676: 71-82

ISSUE DATE:

2010-04

URL:

http://hdl.handle.net/2433/141260

数理解析研究所講究録
第1676巻 2010年 71-82

An Inconsistency Problem in Data Discretion Using Equal Width Interval Approach

Chien-Hsing Wu
Department of Information Management
National University of Kaohsiung
Taiwan
chwu@nuk.edu.tw

Hiroaki Ishii
Graduate School of Information Science and Technology
Osaka University, Japan
ishii@ist.osaka-u.ac.jp

Abstract

Data discretion is one of the most important issues in data mining for continuous datasets. However, there is less attention that has been paid to the inconsistency of dataset that is discreted. In consequence, mining mechanism such as classification may face a difficulty in providing the classification outputs. We first reveal the problem of record inconsistency, and then describe a model that helps efficiently reveal the record inconsistency that do exist datasets that are discreted. The main outputs of the described model include revealed inconsistent records and consumed processing time. Fifteen continuous real-life datasets that are discreted by using the binning technique of S-plus histogram binning algorithm with equal width interval technique are tested. There are results obtained indicating that: (1) 38.89% of the discreted datasets contain inconsistent records and 22.22% of the discreted datasets have more than 20% amount of inconsistent records.

Keywords: Data mining, Discretion, Record consistency

Background

Data mining (DD) is one of the active research domains that is linked to data management, information representation, and machine learning techniques, in particular the volume of data generated increases rapidly [1-9]. If thoroughly review the DD research tasks, there are five major stages: data collection, the collected data preprocessing, pre-processed data mining, outputs collection, and output implementation and evaluation. The data collection basically is to gather real-life (sometimes artificial) data. Pre-processing deals mainly with data refinement and reconstruction of datasets, consistency of multi-typed datasets, elimination of redundant attributes, combination of highly correlative attributes, and

discretization of numeric attributes. The mining mechanism generally performs association, classification, regression, clustering, or summarization to explore knowledge that is significantly interesting, meaningful, and decision-supportable. Outputs contain the discovered knowledge that can be either documented as a report or used in intelligent systems to support in making decision. The work of implementation and evaluation is about the use of mined knowledge and test of the mining processes. It is dealt to ensure the DD pursuits that include reliability, efficiency, validity, simplicity, and/or generality. Obviously, in the DD research system, the results of each phase flow down to the next, and will have a great impact on the final results.

Discretion is the conversion of continuous attributes to categorical ones in order for mining mechanism to perform knowledge discovery. However, conflicting records may occur that have the same conditions, but different conclusion [10]. Although many studies have presented various techniques to improve the performance of discretion, the problem of record inconsistency in a dataset is still a serious issue that may consequently influence the reliability of the mind knowledge [2][3][8][11-18]. It should be noted that indisputably same conditions resulting in different conclusions in our real life is a common situation. However, with respect to an induction-based knowledge discovery mechanism, it should be defined that the same conditions must produce a single conclusion. For example, it will be a meaningless mined knowledge that if you study quite hard, you may or may not pass your final exam. Consequently, the investigation of record consistency for discreted datasets is quite important for data mining. A discreted dataset usually contains many attributes and many records. It is a highly labor-consumption task to investigate the conflicting records. We therefore adopt Structured Quesy Language (SQL) to develop a record consistency investigation model to help efficiently test discreted datasets [19-21]. The primary outputs of the described model include explored inconsistent records and consumed running time.

Record Inconsistency Problem

Before explaining the record inconsistency problem, we briefly describe the discretion. The discretion is to group numeric data for

each attribute in dataset. Figure 1 illustrated this in a graphic manner with equal width interval (EWI). It should be noted that there are many techniques used to group numeric data. However, this is beyond our research focus. The EWI has been often used as a conversion mechanism to generate nominal values from continuous ones. The EWI deals with the sorting the observed values of a continuous attribute and dividing the range of observed values for the variable into k equally sized bins, where k is a parameter predefined by the user. If a variable x is observed to have values bounded by x_{max} and x_{min}, then this method computes the equalized bin width as $(x_{max} - x_{min})/k$. As a result, the set of granules can be expressed as $G = \{G_1, G_2, \cdots\cdots G_k\}$. The conversion function is defined as follows. Each nominal granule has the same continuous boundary, but the number of record may be different.

$$C_{EWI}(x_i) = \begin{cases} G_m, & if\ x_{min} + (m\mp)d \leq x_i \square\ x_{min} + md \\ G_k, & if\ x_i = x_{max} \\ null & otherwise \end{cases}$$

where

x_i: the i^{th} data.

G_m : the m^{th} granule that y is grouped into, m = 1, ···k, k: the number of granules.

x_{min}: the minimum of the data.

x_{max}: the maximum of the data.

d: the equalized interval for k granules.

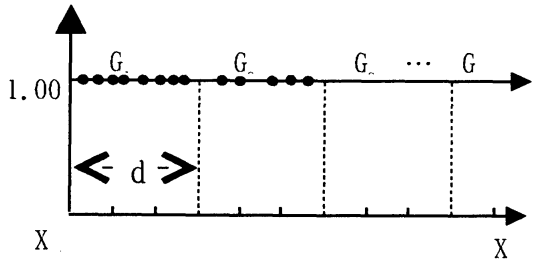

Figure 1: Information granularity

It is very possible that inconsistent records may occur while performing the conversion operation. Conflicting records are very possible to produce dead-end leaf while constructing a decision tree [10]. In other words, conflicting records will stop returning decision rules. It is believed that conflicting records will very possibly result in two serious problems. One is that it is irrelevant to compute gained information for both attributes and classes if a discreted dataset contains too many inconsistent records. The other is that some information of the collected data sources will be lost if too many inconsistent records are eliminated.

To further deal with such a serious problem while discreting the continuous attributes, an artificial dataset is firstly generated as an example that has the record inconsistency problem to describe this concern in detail and secondly conduct an empirical investigation for real life datasets. The generated dataset contains 28 discreted records. Each record is assigned an ID. Also assume that the dataset has 7 attributes as well as a class with 7 labels and is ordered by the values of both attributes and class. The value domain of attributes for this dataset are: V_A={A1, A2, A4, A5, A6, A7}, V_B={B1, B2, B3, B5, B6, B7} V_C={C1, C2, C4, C5, C6, C7}, V_D={D1, D2, D4, D5, D6, D7}, V_E={E1, E2, E3, E4, E5, E6}, V_F ={F2, F3, F4, F5, F6}, V_G={G1, G2, G3, G4, G5, G7}, and V_{CSS}={CSS1, CSS2, CSS3, CSS4, CSS5, CSS6, CSS7}. If look at the records numbered from 1 to 7, it is found that the combinations of attributes are all equal to (A1, B2, C2, D4, E1, F2, G2), but have different conclusions that are {CSS1, CSS2, CSS3, CSS5}.

Since inconsistent records are regarded as those that have the same conditions, but different conclusions, any record that has a single attribute values is considered to be consistent in a discreted dataset. Moreover, in order to further conveniently look at the inconsistency problem in depth and find a solution in general, four types of subdatasets based on the relations between attribute values and class values for a discreted dataset are defined as follows.

$D_{initial}$: A dataset that contains all records in a discreted dataset. For example, $D_{initial}$ = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28} in the artificial dataset. D_{single}: A dataset that contains every single

record of which the combination of attribute value is unique. For example, D_{single} = {8, 9, 10, 18, 23, 27, 28} in the artificial dataset. $D_{multiple}$: A dataset that contains record sets. Each set has an identical combination of attribute value and the number of records is equal to or greater than 2. For example, $D_{multiple}$ = {{1, 2, 3, 4, 5, 6, 7}, {11, 12, 13, 14}, {15, 16, 17}, {19, 20}, {21, 22}, {24, 25, 26}} in the artificial dataset. Note that $D_{multiple}$ is equivalent to the difference set of $D_{initial}$ and D_{single}. $D_{SameConclusion}$: A dataset that contains record sets. Each set is a member of $D_{multiple}$ and has a unique conclusion. For example, $D_{SameClass}$ = {{19, 20}, {24, 25, 26}} in the artificial dataset.

Apparently, the inconsistent records only exist in $D_{multiple}$. Each element (a subset) in $D_{multiple}$ needs to be further detected whether or not the records have the same conclusions. Any element of which the conclusions are constant is eliminated from $D_{multiple}$ to derive the set that contains all inconsistent record. For example, $S_{SameClass}$ containing two record sets of {19, 20} and {24, 25, 26} are such kind of elements in $S_{multiple}$. The remainder then forms a subset that is denoted by $D_{DifClass}$ that contains the inconsistent record which are {{1, 2, 3, 4, 5, 6, 7}, {11, 12, 13, 14}, {15, 16, 17}, {21, 22}}. $D_{DifClass}$ is then used to investigate the initial dataset. However, for the algorithmic aspect, formula (1) is not just a simple algebra that can be used to explore inconsistent records. The size of dataset, number of attributes, number of values that each attribute can take on, and the number of values that the class can take on are all variables that may cause the problem very complex. In order for the solution to be generalized, the study needs to solve two problems. One is to separate D_{single} and $D_{multiple}$ from $D_{original}$, the other is to eliminate all subsets in $D_{SameClass}$ from $D_{multiple}$.

Proposed Model

The proposed model basically is developed by using a divide-group-join strategy. It contains three procedures: grouping, dividing, and joining. Grouping is to deal with the D_{single} and $D_{multiple}$ separations from the initial dataset. The $D_{SameClass}$ is derived by dividing while joining returns the final results. In Figure 2, there are three operations, OP_A, OP_B, and OP_C to complete the whole process. Each operation represents an SQL statement for a defined purpose. The

objective of each operation is listed in Table 1 where operations and their corresponding SQL statements with pseudo format are contained. By connecting all procedures, the defined SQL statements used to return inconsistent records is formed.

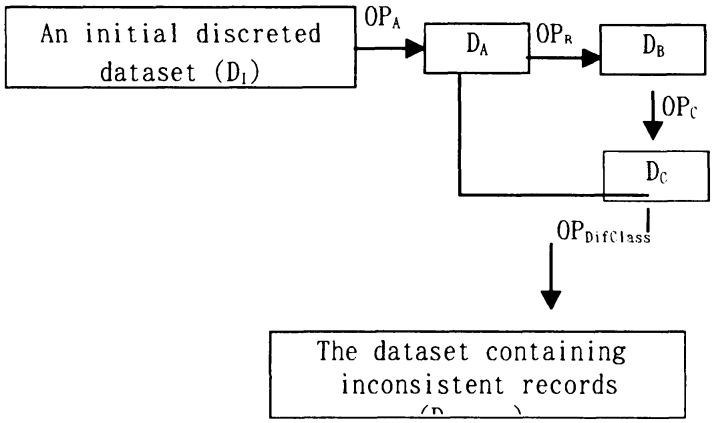

Figure 2: The operations of record consistency

Table 1: The operations and corresponding SQL statements

Operations	SQL Statements (pseudo format)	datasets	Description
OP_A	Select *, Count(dataset.a); From D_I; Group by all_attribute + class; Order by all_attribute + class; Into Table D_A	D_I	The initial discreted dataset
		D_A	The created temporary dataset that lists the number of records of which the conditions and conclusion are constant.
OP_B	Select *, Count(dataset.b); From D_A; Group by all_attribute; Order by all_attribute; Into Table D_B	D_B	The created temporary dataset from D_A that lists all kinds of combinations of conditions and their number of occurrences.
OP_C	Delete From D_B ; Where cnt_b<2	D_C	The created temporary dataset from S_B that lists all combinations of conditions for inconsistent records.
$OP_{DifClass}$	Select * ; From D_A , D_C ; Into Table $S_{DifClass}$; Where D_A.all_attribute = S_C.all_attribute	$D_{DifClass}$	The final subset that contains all inconsistent records.

A demonstrated example

An example as mentioned in the above section is used to demonstrate how the proposed model functions in a simpler way. D_A was listed in Table 2 where the number of records with same conditions and conclusions were contained. In D_A, we realized that some records were consistent and some were not. However, there was only one record in D_A if they were consistent, since all conditions and conclusions were taken into account in terms of grouping. Therefore, we regrouped via all conditions and counted for D_A. The output was stored in D_B that was listed in Table 3. We then eliminated those records from D_B that returned only one record to obtain D_C listed in Table 4, because they represented consistent records in D_A. We finally performed inner join for D_A and D_C to obtain the final results $D_{DifClass}$. The $D_{DifClass}$ was listed in Table 5. All intermediate subsets as well as the final results were tested correct.

Table 2: The subset

ABCDEFGClassCnt_aA1B2C2D4E1F2G2CSS13
A1B2C2D4E1F2G2CSS22A1B2C2D4E1F2G2CSS
31A1B2C2D4E1F2G2CSS51A1B2C2D4E1F2G3C
SS41A1B3C4D4E1F3G3CSS41A1B3C4D4E1F3G
4CSS21A2B3C4D5E2F3G1CSS12A2B3C4D5E2F
3G1CSS52A4B5C5D5E2F4G3CSS11A4B5C5D5E
2F4G3CSS31A4B5C5D5E2F4G3CSS41A4B5C5D
5E3F4G5CSS51A4B5C5D6E3F5G7CSS22A5B1C
5D6E4F5G4CSS41A5B1C5D6E4F5G4CSS61A5B
6C6D6E5F5G3CSS71A6B7C2D7E5F6G2CSS33A
7B3C1D7E6F6G5CSS21A7B7C7D7E6F5G2CSS3
1

Table 3: The subset

ABCDEFGClassCnt_aCnt_bA1B2C2D4E1F2G2CSS514A
1B2C2D4E1F2G3CSS411A1B3C4D4E1F3G3CSS411A1B3
C4D4E1F3G4CSS211A2B3C4D5E2F3G1CSS522A4B5C5D
5E2F4G3CSS413A4B5C5D5E3F4G5CSS511A4B5C5D6E3
F5G7CSS221A5B1C5D6E4F5G4CSS612A5B6C6D6E5F5G
3CSS711A6B7C2D7E5F6G2CSS331A7B3C1D7E6F6G5CS
S211A7B7C7D7E6F5G2CSS311

Table 4: The subset

ABCDEFGClassCnt_aCnt_bA1B2C2D4E1F2G2CSS514A
2B3C4D5E2F3G1CSS522A4B5C5D5E2F4G3CSS413A5B1
C5D6E4F5G4CSS612

Table 5: The subset

ABCDEFGClassCnt_aA1B2C2D4E1F2G2CSS13
A1B2C2D4E1F2G2CSS22A1B2C2D4E1F2G2CSS
31A1B2C2D4E1F2G2CSS51A2B3C4D5E2F3G1C
SS12A2B3C4D5E2F3G1CSS52A4B5C5D5E2F4G
3CSS11A4B5C5D5E2F4G3CSS31A4B5C5D5E2F
4G3CSS41A5B1C5D6E4F5G4CSS41A5B1C5D6E
4F5G4CSS61

Experiment and Results

In order for the proposed model to be able to both investigate the record inconsistency and reveal record inconsistency problem, 18 continuous real-life datasets were tested. The discretion technique employed was the EWI that involves sorting the observed values of a continuous feature and dividing it into k equally sized granules, where k is the number of granules defined by users [2]. The binning algorithm, S-Plus Histogram Binning Algorithm (SHBA) introduced by Spector [22] was utilized to determine the number of granules. The characteristics of the experiment were that (1) the number of datasets used is 18, (2) missing data was eliminated if occurs, (3) binning technique utilized is SHBA, (4) discretion technique employed is EWI, (5) outputs of the experiment is inconsistent records and processing time, and (6) objective is the investigation of record

inconsistency. The SHBA is expressed as SHBA: k = maximum(1, integer(2*log(difference))), where k is the number of granules and difference is the number of different values that an attribute has.

In addition, the average of run time was the mean of twenty trials that were performed. The results of the experiment were listed in Table 7. It was found that of 18 datasets that were investigated, 7 were not recognized. This implied that 38.89% of the discreted datasets contained inconsistent records. More particularly, there was four datasets that contained inconsistent records more than 20%. The one that showed the biggest percentage of inconsistent records was 37.2373%. The experimental results also confirmed that the proposed model did not consume unacceptable processing time.

Table 7: Record consistency investigation via EWI with SHBA

Datasets	Size	Num. of attributes	Num. of class	Pct. of record inconsistency	Avg. of run time (seconds)
BC198	194	33	2	0.0000	0.3040
BC569	569	30	2	0.0000	0.4680
Bupa	345	6	2	28.9855	0.2220
Sonar	208	60	2	0.0000	0.5910
Satellite	2000	36	6	0.0000	1.9900
Pageblock	5473	10	5	37.2373	0.5010
Letter	16384	16	26	3.0945	6.0260
Pendigit	3498	16	10	0.0000	1.2430
BC699	699	9	2	0.5722	0.2530
Glass	214	9	7	19.1589	0.1480
Iris	150	4	3	4.0000	0.1220
Segmentation	210	16	7	0.0000	0.2320
Shuttle	14500	9	7	34.0828	2.2100
Synthetic	600	61	6	0.0000	0.8900
Vehicle	846	18	4	0.0000	0.4610
Vowel	900	10	11	0.0000	0.3560
Waveform	5000	21	3	0.0000	2.1490
Wine	178	13	3	0.0000	0.1880

Conclusion

This paper has briefly described the problem of record inconsistency and presented a model that is constructed by using SQL to help efficiently investigate the inconsistent records for discreted datasets. It has been demonstrated successful as a record inconsistency detector for any dataset

that is discreted. The model can be also directly used for discrete datasets. The equal width interval technique with SHBA embedded is employed to in the proposed model. The results implied that record inconsistency investigation was truly an essential issue to the discreted datasets used in DM research. In our experiment, many parameters are considered. For example, the determination of k is based on Spector [22]. Other techniques may have influence the final results, which is one of our future research focuses. Although that we believe that other discretion techniques utilized and binning regulations employed may greatly affect the percentage of inconsistent records in a discreted dataset, it is encouraged that datasets used in the DM research be investigated before moving to the next stages. Moreover, as we have mentioned, the decision of the size of granules for continuous attributes that need to be discreted is facing a problem of dilemma that suffers from unacceptable amount of conflicting records to be contained if too large and incomprehensible knowledge to be discovered if too small. Therefore, a mechanism that can carry out a near optimal solution for record consistency and reliability of a descreted rule also is advantageous to the further research focuses.

References

[1] R. Agrawal, T. Imielinski, A. Swami, Database Mining: A Performance Perspective, IEEE Trans. on Knowledge and Data Eng., 5(1993), 914-925.

[2] J. Dougherty, R. Kohavi, M. Sahami, Supervised and Unsupervised Discretization of Continuous Features, A. Prieditis & S. Russell (Ed.), Proceedings of 1995 International Conference on Machine Learning, CA: Morgan Kaufmann, 1995, 194-202.

[3] B. Pfahringer, Compression-Based Discretization of Continuous Attributes, Proceedings of the 12th International Conference on Machine Learning, CA: Morgan Kaufmann, 1995, 456-463.

[4] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, The KDD Process for Extracting Useful Knowledge form Volumes of Data, Communication ACM, 39(1996), 27-41.

[5] T. Imielinski, H. Mannila, A Database Perspective on Knowledge Discovery, Communication of ACM, 39(1996), 58-64.

[6] M.S. Chen, J. Han, P.S. Yu, Data Mining: An Overview from a Database Perspective, IEEE Transaction on Knowledge and Data Engineering, 8(1996), 866-883.

[7] U.M. Fayyad, P. Stolorz, Data Mining and KDD: Promise and

Challenges, Future Generation Computer Systems, 13(1997), 99-115.

[8] M. Pazzani, S. Mani, W.R. Shankle, Comprehensible Knowledge-Discovery in Databases. M. G. Shafto and P. Langley (Ed.), Proceedings of The Nineteenth Annual Conference of The Cognitive Science Society, Hilldale NJ: Lawrence Erlbaum, 1997, 596-601.

[9] K. Hirota, W. Pedrycz, Fuzzy Computing for Data Mining, Proceedings of The IEEE, 87(1999), 1575-1600.

[10] Sestito, S. & Dillon, T. Automated Knowledge Acquisition, NJ: Prentice Hall 1994.

[11] C.C. Chan, C. Batur, J.W. Srinivasasn, Determination of Quantization Intervals in Rule Based Model for Dynamic Systems, Proceedings of The IEEE Conference on Systems, Man, and Cybernetics, Charlottesville, VA., 1991 1719-1723.

[12] R. Kerber, ChiMerge: Discretization of Numeric Attributes, Proceedings of the 10th National Conference on Artificial Intelligence, Menlo Park: MIT Press, 1992, 123-128.

[13] J. Catlett, On Changing Continuous Attributes into Ordered Discrete Attributes, Proceedings of the European Working Session on Learning, Berlin, Germany: Springer-Verlag, 1991, 164-178.

[14] U.M. Fayyad, K.R. Irani, Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning, Proceedings for the 13th International Joint Conference on Artificial Intelligence, CA: Morgan Kaufmann, 1993, 1022-1027.

[15] M.R. Chmielewski, J.W. Grzymala-Busse, Global Discretization of Continuous Attributes as Preprocessing for Machine Learning, International Journal of Approximating Reasoning, 15 (1996), 319-331.

[16] M. Pazzani, An Iterative-Improvement Approach for The Discretization of Numeric Attributes in Bayesian Classifiers,

Proceedings of the First International Conference on Knowledge Discovery and Data Mining, Montreal: AAAI Press, 1995.

[17] H. Liu, R. Setiono, Feature Selection Via Discretization, IEEE Transaction. Knowledge and Data Engineering, 9(1997), 642-646.

[18] X. Wu, D. Urpani, Induction By Attribute Elimination, IEEE Transaction. on Knowledge and Data Engineering, 11(1999), 805-812.

[19] J. Han, Y. Fu, K. Koperski, O. Zaiane, DMQL: A Data Mning Query Language for relational Databases, DMKD-96 (SIGMOD-96 Workshop on KDD), Montreal, Canada, 1996.

[20] R. Meo, G. Psaila, S. Ceri, An Extension to SQL for Mining Association Rules, Data Mining & Knowledge Discovery, 2(1998), 195-224.

[21] T. Imielinski, A. Virmani, MSQL: A Query language for Database Mining, Data Mining & Knowledge Discovery, 3(1999), 373-408.

[22] P. Spector, An Introduction To S and S-Plus, Belmont CA: Duxburg Press, 1994.